

MBASE: Model-Based Architecting and Systems/Software Engineering (CS577a-b Version; full theory later in course)

CS 577a

Software Engineering I

Barry Boehm, USC

Dan Port, USC

David Klappholz, Stevens

Fall 2002

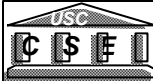


**“No scene from prehistory is quite so vivid as that of the
mortal struggles of great beasts in the tar pits.**



**Large system programming has over the past decade been such
a tar pit, and many great and powerful beasts have thrashed
violently in it.”**

Fred Brooks, 1975



“Everyone seems to have been surprised by the stickiness of the problem, and it is hard to discern the nature of it.



But we must try to understand it if we are to solve it.”

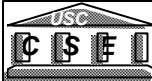
Fred Brooks, 1975



Understanding the Tar Pit: Model Clashes



- **Model Clash: An incompatibility among the underlying assumptions of a set of models**
 - Process, product, property, and success models
 - Often unrecognized
 - Produces conflicts, confusion, mistrust, frustration, rework, throwaway systems



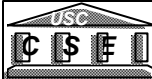
Where CS577a-b Version of MBASE Comes From

- Waterfall (Requirements, Design, Implementation, Testing)
- Spiral
- Risk-Driven Spiral
- WinWin Risk-Driven Spiral
- WinWin Risk Driven Spiral with Anchor Points (LCO, LCA, IOC)
- MBASE: WinWin Risk-Driven Spiral with model conflict resolution



Anchor Points

- Life Cycle Objectives (LCO): Stakeholders agree on system's requirements and developers have shown one feasible architecture...like getting engaged (B.B.)
- Life Cycle Architecture (LCA): Developers have detailed an implementation architecture and stakeholders agree to proceed to implementation...like getting married (B.B.)
- Initial Operating Capability (IOC): Working system has been implemented, tested, and transitioned to customer site, and stakeholders have agreed to put it into operation... like having a first baby (B.B.)



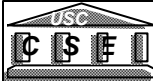
Model Clashes in Overdue Book Notice System

- **(software) model:** a consistent set of assumptions made by a stakeholder about a software development project
- **model clash:** incompatibility between assumptions in different models
- **assumption:** a statement that is taken for granted by a stakeholder (whether or not the stakeholder is consciously aware that s/he is doing so)



Categories of Models

- **Product model:** consistent set of assumptions about system and related elements
- **Property model:** consistent set of assumptions about value of an attribute of system or project
- **Process model:** consistent set of assumptions about project activities whereby system is developed
- **Success model:** consistent set of assumptions about criteria for stakeholder satisfaction



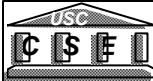
Library Overdue Book Notices I

- Do notices look exactly like they always have?
- Teachers are conservative and will resist changing their routine in any way
- **Possible Model Clashes:**
 - **Teachers makes the following assumption:**
 - **success model assumption: no changes should be made to procedure or forms**
 - **Developer makes one or both of following assumptions:**
 - **product model assumption: fancier graphics make product more appealing**
 - **process model assumption: stakeholder interaction is not necessary for requirements gathering**



Library Overdue Book Notices II

- Is output sorted by class, and then by student name?
- Librarian gives notices to teachers to give to students. If s/he has to sort by hand, s/he won't use the system.
- **Possible Model Clashes:**
 - **Librarian makes the following assumption:**
 - **success model assumption: system should result in less work rather than more work**
 - **Developer makes one or both of following assumptions:**
 - **product model assumption: output order is unimportant**
 - **process model assumption: stakeholder interaction is not necessary for requirements gathering**



Library Overdue Book Notices III

- Is input by books NOT returned, rather than by books returned?
- Far more books are returned on time than not.
- Far more work to enter books returned.
- **Possible Model Clashes:**
 - **Librarian makes following assumption**
 - **success model assumption: system should result in less work rather than more work**
 - **Developer makes one of the following assumptions:**
 - **product model assumption: either input mode is as good as the other**
 - **process model assumption: stakeholder interaction is not necessary for requirements gathering**

8/30/02

©USC-CSE

11



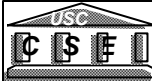
Library Overdue Book Notices IV

- Does the manual make it clear that data must be saved to floppy after each time the system is used?
- If not, computer-shy librarian may forget.
- **Possible Model Clashes:**
 - **Librarian makes the following assumption”**
 - **success model assumption: all required actions must be stated explicitly in manual**
 - **Developer makes one, two, or three of the following assumptions:**
 - **success model assumption: users are as sophisticated as I am**
 - **process model assumption: stakeholder interaction is not necessary for requirements gathering**
 - **success model assumption: expend least possible effort**

8/30/02

©USC-CSE

12



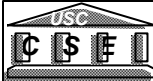
Library Overdue Book Notices V

- Is saving done via a menu command?
- Teaching librarian to copy files is much harder than teaching him/her to use a menu command.
- **Possible Model Clashes:**
 - Librarian makes one or both of following assumptions:
 - success model assumption: the simpler the better
 - product model assumption: “save” function is part of GUI
 - Developer makes one or both of the following assumptions
 - property (environment) model assumption: users are sophisticated
 - process model assumption: stakeholder interaction is not necessary for requirements gathering



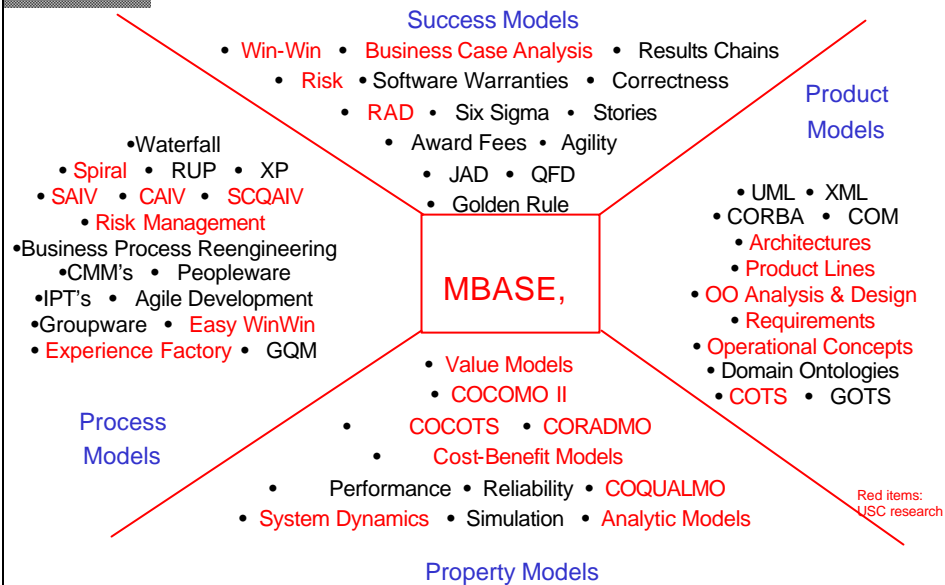
Library Overdue Book Notices VI

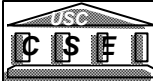
- Does system prompt for save upon exiting?
- If not, librarian may very well forget.
- **Possible Model Clashes**
 - Librarian makes one or both of the following assumptions:
 - success model assumption: I can't afford to ever lose data
 - product model assumption: “save” function should be prompted
 - Developers make one or both of the following assumptions
 - property (environment) model assumption: users are sophisticated
 - process model assumption: stakeholder interaction is not necessary for requirements gathering



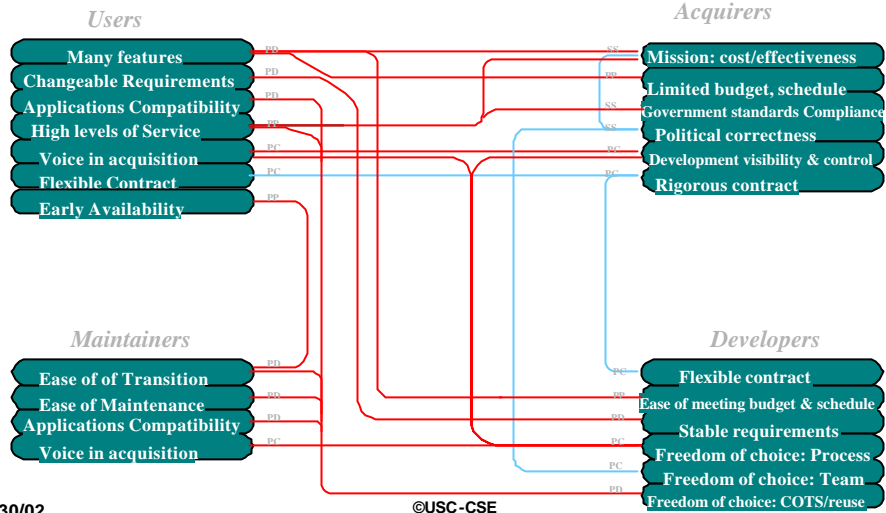
Library Overdue Book Notices VII

- Did you add features/functionality to impress the customer?
- What makes you think they will impress?
- Have you asked the customer if s/he's willing to pay the extra cost?
- **Possible Model clash:**
 - Librarian makes following assumption:
 - success model assumption: I should be able to learn entire system in an afternoon
 - AND
 - EITHER Customer makes following assumption:
 - success model assumption: I don't want to be charged for things I didn't ask for
 - OR Developer makes the following three success model assumptions:
 - I want to maximize profits
 - the more features the more customers
 - the more customers the more profits





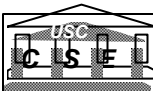
Success Model-Clashes profile MasterNet



8/30/02

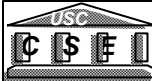
©USC-CSE

17



**Model Clashes are
Resolved/Avoided by Following
MBASE Guidelines
Which Provide Guidance for
Producing and Documenting
Models**

[http://sunset.usc.edu/classes/
cs577a_2001/guidelines/MBASE
_Guidelines_v2.3.3.pdf](http://sunset.usc.edu/classes/cs577a_2001/guidelines/MBASE_Guidelines_v2.3.3.pdf)



MBASE: Creating and using models to develop a product


Documents (artifacts) are produced to communicate the models to stakeholders (not for their own sake)


Every model should serve a purpose; if not, do not produce or document it

Models/documentation can change over the course of a project


MBASE guidelines are guidance, not gospel



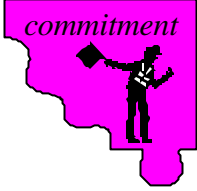
-  • **Overview of MBASE Models/Documents**
- Operation Concept Description (OCD)
 - System and Software Requirements Definition (SSRD)
 - System and Software Architecture Description (SSAD)
 - Life Cycle Plan (LCP)
 - Feasibility Rationale Description (FRD)


 University of Southern California
 Center for Software Engineering

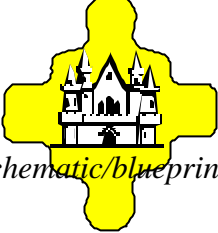
Ingredients of a project




concept



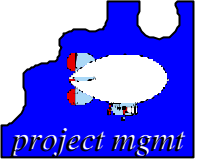
commitment




schematic/blueprint



plan




project mgmt




Product Const.
Delivery & Maint.


8/30/02 ©USC-CSE 21


 University of Southern California
 Center for Software Engineering


MBASE Artifact Collections



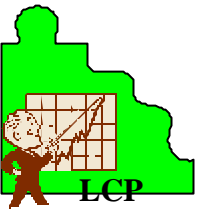
OCD



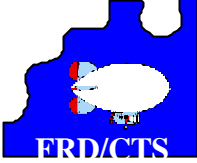
SSRD




SSAD



LCP

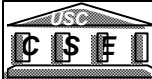


FRD/CTS

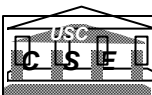


CTS

8/30/02 ©USC-CSE 22

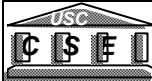


- Overview of MBASE
- ➔ **Operation Concept Description (OCD)**
- System and Software Requirements Definition (SSRD)
- System and Software Architecture Description (SSAD)
- Life Cycle Plan (LCP)
- Feasibility Rationale Description (FRD)



Purpose of OCD

- **Describe context of system. What do we have now? why build a new system?**
- **Describe stakeholders of the system: how the system will work when deployed.**
- **Describe evolution from current to new operational concept. Clarify the value of new system.**



- Overview of MBASE
- Operation Concept Description (OCD)
- ➔ • **System and Software Requirements Definition (SSRD)**
- System and Software Architecture Description (SSAD)
- Life Cycle Plan (LCP)
- Feasibility Rationale Description (FRD)

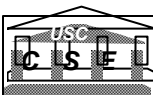


Purpose of SSRD

- **Describe Capability Requirements: system subject matter measured by concrete means**
- **Describe Project, Level of Service, and Evolutionary (non-functional) Requirements: Behavioral properties of functions (performance, usability, etc.)**
- **Describe Global Constraints: constraints that apply to the system as a whole (including interface, environment and Data reqs)**
- **Mandates (“must”, “shall”, “will”): how the system must be implemented with respect to general tech.**

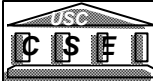


- Overview of MBASE
- Operation Concept Description (OCD)
- System and Software Requirements Definition (SSRD)
- ➔ • **System and Software Architecture Description (SSAD)**
- Life Cycle Plan (LCP)
- Feasibility Rationale Description (FRD)



Purpose of SSAD

- Documents the Architectural Analysis and the System Design, “blueprint”
- Serves As A Bridge between Engineering (Inception and Elaboration) and Construction Phase
- The SSAD is refined during the Construction Phase into a Software Detailed Design Specification
- The SSAD should not repeat information from other documents: reference other information. *Vital to project integration and cohesion. Conciseness is paramount*



- Overview of MBASE
- Operation Concept Description (OCD)
- System and Software Requirements Definition (SSRD)
- System and Software Architecture Description (SSAD)
- ➔ • **Life Cycle Plan (LCP)**
- Feasibility Rationale Description (FRD)

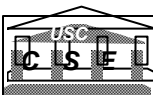


Purpose of LCP

- Basis for monitoring and controlling the project's progress
- Basis for controlling the project's progress in achieving the software product objectives
- Helps make the best use of people and resources throughout the life cycle
- Provides evidence that the developers have thought through the major life cycle issues in advance
- Organized to answer the most common questions about a project or activity: why, what, when, who, where, how, how much, and whereas
- Maintaining the conceptual integrity between the tasks and the things they create/solve is a prime quality criterion.



- Overview of MBASE
- Operation Concept Description (OCD)
- System and Software Requirements Definition (SSRD)
- System and Software Architecture Description (SSAD)
- Life Cycle Plan (LCP)
- ➔ • **Feasibility Rationale Description (FRD)**



Purpose of FRD

- Ensure feasibility and consistency of other package components (OCD, SSRD, SSAD, LCP, Prototype)
- Demonstrate viable business case for the system
- Identify shortfalls in ensuring feasibility, consistency, and business case as project risk items for LCP
- Demonstrate that a system built using the specified architecture (described in the SSAD) and life cycle process (described in the LCP) will fulfill prediction of other docs
- Rationalize life cycle decisions in a way the prime audience (the customer and users) and other stakeholders can understand
- Enable the customers to participate in the decision process and to express their satisfaction with the product